

Unidad 3- Manipulando strings

Comillas simples y comillas dobles

A la hora de trabajar con strings podremos usar comillas dobles o simples. La gran diferencia radica en que las comillas dobles toman en cuenta las acciones de los caracteres especiales y las comillas simples no.

Veamos el siguiente ejemplo:

```
<?php
    $nombre = "Christian";

    echo "Hola $nombre";
    echo 'Hola $nombre';
```

En el primer caso se despliega "Hola Christian" y en el segundo caso despliega el mensaje "Hola \$nombre"

Tamaño de un string

Para determinar el tamaño de un string utilizaremos la función strlen()

```
<?php
    $nombre = "Christian";

    echo strlen($nombre);
```

Esta función mostrará el número de caracteres de la variable nombre, que en este caso es 9 (también se le puede pasar directamente una cadena como argumento a strlen en lugar de la variable)

Conteo de palabras

Para realizar un conteo de palabras usaremos la función `str_word_count()`. En el siguiente caso se mostrará el valor 3

```
<?php
    $nombre = "Buenos dias familia";

    echo str_word_count($nombre);
```

Buscar un texto dentro de un string

A diferencia de las funciones anteriores, en la función de búsqueda requerimos el ingreso de dos argumentos: el string a analizar y el texto a buscar. Literalmente es como buscar una aguja en un pajar (según la documentación de PHP). Si el texto se encuentra, mostrará la posición en la que se encuentra la primera aparición del texto. En caso contrario retornará **false**.

```
<?php
    $cadena = "Buenos dias familia";

    echo "Valor: ".strpos($cadena, "nos");
    echo "<br>";
    var_dump(strpos($cadena, "auto"));
```

En el primer caso nos muestra el valor 3 (ya que los string comienzan a contar a partir de la posición 0 al ser un dato de tipo arreglo que veremos más adelante)

En el segundo caso, utilizamos la función `var_dump` para identificar el valor de retorno de la función (en este caso nos muestra `bool(false)`)

Cambiar un string a mayúsculas o minúsculas

Para realizar esta acción debemos usar la función `strtoupper()`

```
<?php
    $cadena = "Buenos dias familia";

    echo strtoupper($cadena);
```

En el caso de pasar todo a minúsculas usamos la función `strtolower()`

```
<?php
    $cadena = "Buenos dias familia";

    echo strtolower($cadena);
```

Reemplazar una subcadena de texto dentro de un string

Para reemplazar una subcadena de texto usaremos la función `str_replace()` en la cual necesitamos ingresar el texto a reemplazar, el nuevo texto y el string en donde hay que reemplazar. En el ejemplo anterior se mostrará la cadena "Buenos días alumnos"

```
<?php
    $cadena = "Buenos dias familia";

    echo str_replace("familia", "alumnos", $cadena);
```

Invertir una cadena de texto

Para invertir una cadena de texto requerimos de un string y utilizar la función `strrev()`. Nos mostrará la cadena en formato inverso.

```
<?php
    $cadena = "Buenos dias familia";

    echo strrev($cadena);
```

Extraer los espacios en blanco de una cadena del inicio y final del mismo

En ocasiones tendremos que remover los espacios en blanco que nos quedan en cadenas de caracteres al inicio y al final de la cadena. Para ello usaremos la función `trim()`. El siguiente ejemplo mostrará la cadena "Buenos días familia" sin los espacios en blanco al inicio y al final

```
<?php
    $cadena = " Buenos dias familia ";

    echo trim($cadena);
```

Convertir cadena en arreglo de datos

Para separar una cadena de caracteres en varios elementos dentro de una colección usaremos la función `explode()`. Los argumentos que espera recibir son un separador y una cadena de texto a separar. Para mostrar dicha cadena usaremos la función reservada `print_r()`

```
<?php
    $cadena = "Buenos dias familia";
    $coleccion = explode(" ", $cadena);

    print_r($coleccion);
```

Array ([0] => Buenos [1] => dias [2] => familia)

Ya veremos que es un Array en la siguiente unidad